

```

BBBBBBBBBBBBBB      AAAAAAAAAA      SSSSSSSSSSSSSS      RRRRRRRRRRRR      TTTTTTTTTTTTTTTT      LLL
BBBBBBBBBBBBBB      AAAAAAAAAA      SSSSSSSSSSSSSS      RRRRRRRRRRRR      TTTTTTTTTTTTTTTT      LLL
BBBBBBBBBBBBBB      AAAAAAAAAA      SSSSSSSSSSSSSS      RRRRRRRRRRRR      TTTTTTTTTTTTTTTT      LLL
BBB      BBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAAAAAAAAAAAAAAAAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAAAAAAAAAAAAAAAAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAAAAAAAAAAAAAAAAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBB      BBB      AAA      AAA      SSS      SSS      RRR      RRR      TTT      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSSSSSS      RRR      RRR      TTT      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSSSSSS      RRR      RRR      TTT      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSSSSSS      RRR      RRR      TTT      TTT      LLL

```

```
BBBBBBBBB      AAAAAA      SSSSSSSS      DDDDDDDD      IIIIII      SSSSSSSS      PPPPPPPP      AAAAAA      TTTTTTTTTT
BBBBBBBBB      AAAAAA      SSSSSSSS      DDDDDDDD      IIIIII      SSSSSSSS      PPPPPPPP      AAAAAA      TTTTTTTTTT
BB      BB      AA      AA      SS      DD      DD      II      II      SS      PP      PP      AA      AA      TT
BB      BB      AA      AA      SS      DD      DD      II      II      SS      PP      PP      AA      AA      TT
BB      BB      AA      AA      SS      DD      DD      II      II      SS      PP      PP      AA      AA      TT
BB      BB      AA      AA      SS      DD      DD      II      II      SS      PP      PP      AA      AA      TT
BBBBBBBBB      AA      AA      SSSSSS      DD      DD      II      II      SSSSSS      PPPPPPPP      AA      AA      TT
BBBBBBBBB      AA      AA      SSSSSS      DD      DD      II      II      SSSSSS      PPPPPPPP      AA      AA      TT
BB      BB      AAAAAAAAAA      SS      DD      DD      II      II      SS      PP      AAAAAAAAAA      TT
BB      BB      AAAAAAAAAA      SS      DD      DD      II      II      SS      PP      AAAAAAAAAA      TT
BB      BB      AA      AA      SS      DD      DD      II      II      SS      PP      AA      AA      TT
BB      BB      AA      AA      SSSSSS      DD      DD      IIIIII      SSSSSS      PP      AA      AA      TT
BBBBBBBBB      AA      AA      SSSSSS      DDDDDDDD      IIIIII      SSSSSS      PP      AA      AA      TT
BBBBBBBBB      AA      AA      SSSSSS      DDDDDDDD      IIIIII      SSSSSS      PP      AA      AA      TT
```

....  
....  
....  
....

```
LL      IIIIII      SSSSSSSS
LL      IIIIII      SSSSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SSSSSS
LL      II      SSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLL      IIIIII      SSSSSSSS
```

```
1 0001 0 MODULE BAS$$DISPATCH_T (
2 0002 0 IDENT = '1-021'
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
10 0010 1 * ALL RIGHTS RESERVED.
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
17 0017 1 * TRANSFERRED.
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
21 0021 1 * CORPORATION.
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: BASIC I/O
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 This module contains the Global dispatch tables for the UDF (user data
36 0036 1 formatter) level and REC (record) level for BASIC.
37 0037 1 In addition it contains a routine which signals errors for invalid
38 0038 1 statement types.
39 0039 1
40 0040 1 ENVIRONMENT:
41 0041 1
42 0042 1 AST reentrant - all OWN storage is read only
43 0043 1
44 0044 1 AUTHOR: Donald G. Petersen , CREATION DATE: 07-Dec-78
45 0045 1
46 0046 1 MODIFIED BY:
47 0047 1
48 0048 1 DGP,06-Dec-78 : VERSION 1-001
49 0049 1 1-001 - original. DGP 06-Dec-78
50 0050 1 1-002 - Add some functionality to OTSS$$SIGDIS_ERR. DGP 08-Dec-78
51 0051 1 1-003 - Change dispatch tables to longwords. DGP 11-Dec-78
52 0052 1 1-004 - Add Basic READ to dispatch tables. DGP 12-Dec-78
53 0053 1 1-005 - Change FORLNK require file to OTSLNK. JBS 22-DEC-78
54 0054 1 1-006 - Signal the proper errors in the error routine. DGP 18-Jan-79
55 0055 1 1-007 - Change file name to OTSDISPAT to agree with RTL standards
56 0056 1 and internal comments. JBS 27-JAN-1979
57 0057 1 1-008 - Use 32-bit addresses for externals. JBS 27-JAN-1979
```

! I/O dispatch tables for all languages  
! File: BASDISPAT.B32 Edit: PLL1021



BAS\$\$DISPATCH\_T  
1-021

E 15  
16-Sep-1984 00:19:30  
14-Sep-1984 11:54:52

VAX-11 Bliss-32 V4.0-742  
[BASRTL.SRC]BASDISPAT.B32;1

Page 2  
(1)

```
.. 58      0058 1 1-009 - Track SBL's changes to the statement types in the ISB.
.. 59      0059 1      JBS 09-FEB-1979
.. 60      0060 1 1-010 - Add GET and PUT. DGP 19-Feb-79
.. 61      0061 1 1-011 - Add PRINT USING and straighten up a lot of Basic stuff. DGP
.. 62      0062 1      15-May-79
.. 63      0063 1 1-012 - Add MAT INPUT. DGP 05-Jun-79
.. 64      0064 1 1-013 - Add MAT PRINT. DGP 15-Jun-79
.. 65      0065 1 1-014 - Add remaining FORTRAN statement types. Indexed REWRITE,
.. 66      0066 1      keyed READ, internal READ and WRITE. SBL 18-Jun-1979
.. 67      0067 1 1-015 - Remove all of the FORTRAN entries. They are moved to
.. 68      0068 1      FOR$$DISPATCH_T. Rename this module BAS$$DISPATCH_T.
.. 69      0069 1      JBS 26-JUN-1979
.. 70      0070 1 1-016 - Use ISB symbols for table length. JBS 12-JUL-1979
.. 71      0071 1 1-017 - Add BAS$$SIGDIS JSB. JBS 01-AUG-1979
.. 72      0072 1 1-018 - Add Basic MAT READ and MAT LINPUT. DGP 11-Oct-79
.. 73      0073 1 1-019 - Add REC9 for Mat Print. DGP 12-Oct-79
.. 74      0074 1 1-020 - Add BAS$$REC.MLI1. DGP 12-Oct-79
.. 75      0075 1 1-021 - Add Basic GET by RFA and FIND by RFA. PLL 4-Jun-1982
.. 76      0076 1 --
.. 77      0077 1
.. 78      0078 1 !<BLF/PAGE>
```

```
80 0079 1 |
81 0080 1 | SWITCHES:
82 0081 1 |
83 0082 1 |
84 0083 1 | SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
85 0084 1 |
86 0085 1 |
87 0086 1 | LINKAGE
88 0087 1 |
89 0088 1 |
90 0089 1 | REQUIRE 'RTLIN:OTSLNK';           ! Define all linkages
91 0518 1 |
92 0519 1 |
93 0520 1 | TABLE OF CONTENTS:
94 0521 1 |
95 0522 1 |
96 0523 1 | FORWARD ROUTINE
97 0524 1 |     BAS$$SIGDIS_ERR : CALL_CCB NOVALUE,       ! Signal a dispatch error
98 0525 1 |     BAS$$SIGDIS_JSB : JSB_ODFO NOVALUE;       ! (JSB entry point)
99 0526 1 |
100 0527 1 |
101 0528 1 | INCLUDE FILES:
102 0529 1 |
103 0530 1 |
104 0531 1 | REQUIRE 'RTLIN:RTLPSECT';
105 0626 1 |
106 0627 1 | REQUIRE 'RTLML:OTSISB';           ! Define ISB offsets
107 0795 1 |
108 0796 1 | REQUIRE 'RTLML:OTSLUB';           ! Define LUB offsets
109 0936 1 |
110 0937 1 |
111 0938 1 | MACROS:
112 0939 1 |
113 0940 1 |     NONE
114 0941 1 |
115 0942 1 | EQUATED SYMBOLS:
116 0943 1 |
117 0944 1 |     NONE
118 0945 1 |
119 0946 1 | EXTERNAL REFERENCES:
120 0947 1 |
121 0948 1 |
122 0949 1 | EXTERNAL LITERAL
123 0950 1 |     OTSS_FATINTERR,
124 0951 1 |     OTSS_IO_CONCLO;
125 0952 1 |
126 0953 1 | EXTERNAL ROUTINE
127 0954 1 |     LIB$STOP : NOVALUE;           ! Signal an error and stop
128 0955 1 |
129 0956 1 | +
130 0957 1 | | Formatting level of abstraction
131 0958 1 | |
132 0959 1 |
133 0960 1 | EXTERNAL ROUTINE
134 0961 1 |     BAS$$UDF_RLO : JSB_UDFO NOVALUE,       ! Initialize read list-directed
135 0962 1 |     BAS$$UDF_RL1 : CALL_CCB,               ! Transmit one I/O list element
136 0963 1 |     BAS$$UDF_RL9 : JSB_ODF9 NOVALUE,       ! Terminate read list directed
```

```

137 0964 1 BASS$UDF_WF0 : CALL_CCB NOVALUE,      ! Initialize write formatted
138 0965 1 BASS$UDF_WF1 : CALL_CCB NOVALUE,      ! Transmit one I/O list element
139 0966 1 BASS$UDF_WF9 : JSB_UDF9 NOVALUE,       ! Terminate write formatted
140 0967 1 BASS$UDF_WL0 : JSB_UDF0 NOVALUE,       ! Initialize write list-directed
141 0968 1 BASS$UDF_WL1 : CALL_CCB NOVALUE,       ! Transmit one I/O list element
142 0969 1 BASS$UDF_WL9 : JSB_UDF9 NOVALUE,       ! Terminate write list-directed
143 0970 1 BASS$UDF_RMF0 : JSB_UDF0 NOVALUE,       ! Initialize read memory formatted
144 0971 1 BASS$UDF_RMF1 : CALL_CCB NOVALUE,       ! Transmit one I/O list element
145 0972 1 BASS$UDF_RMF9 : JSB_UDF9 NOVALUE;      ! Terminate read memory formatted
146 0973 1
147 0974 1 !+
148 0975 1 ! Record processing level of abstraction
149 0976 1 !-
150 0977 1
151 0978 1 EXTERNAL ROUTINE
152 0979 1 BASS$REC_RSL0 : JSB_REC0 NOVALUE,      ! Initialize read list-directed
153 0980 1 BASS$REC_RSL1 : JSB_REC1 NOVALUE,      ! Transmit one record
154 0981 1 BASS$REC_RSL9 : JSB_REC9 NOVALUE,      ! Terminate read list-directed
155 0982 1 BASS$REC_WF0 : JSB_REC0 NOVALUE,       ! Initialize write formatted
156 0983 1 BASS$REC_WF1 : JSB_REC1 NOVALUE,       ! Write one record
157 0984 1 BASS$REC_WF9 : JSB_REC9 NOVALUE,       ! Terminate write formatted
158 0985 1 BASS$REC_WSL0 : JSB_REC0 NOVALUE,       ! Initialize write list-directed
159 0986 1 BASS$REC_WSL1 : JSB_REC1 NOVALUE,       ! Write one record
160 0987 1 BASS$REC_WSL9 : JSB_REC9 NOVALUE,       ! Terminate write list-directed
161 0988 1 BASS$REC_RMF0 : JSB_REC0 NOVALUE,       ! Initialize read memory formatted
162 0989 1 BASS$REC_RMF1 : JSB_REC1 NOVALUE,       ! More initialization
163 0990 1 BASS$REC_RMF9 : JSB_REC9 NOVALUE,       ! Terminate read memory formatted
164 0991 1 BASS$REC_GSE : JSB_REC9 NOVALUE,       ! GET sequential (RMS) one record
165 0992 1 BASS$REC_PSE : JSB_REC9 NOVALUE,       ! PUT sequential (RMS) one record
166 0993 1 BASS$REC_MPR9 : JSB_REC9 NOVALUE,       ! terminate Mat Print
167 0994 1 BASS$REC_MIN0 : JSB_REC0 NOVALUE,       ! initialize MAT INPUT
168 0995 1 BASS$REC_MIN1 : JSB_REC1,              ! read one record
169 0996 1 BASS$REC_MIN9 : JSB_REC9 NOVALUE,       ! terminate MAT INPUT
170 0997 1 BASS$REC_MLI1 : JSB_REC1,              ! Mat Linput - read one record
171 0998 1 BASS$REC_MRE1 : JSB_REC1;              ! return failure - MAT READ
172 0999 1
173 1000 1 !
174 1001 1 ! OWN STORAGE:
175 1002 1 !
176 1003 1 DISPATCH_PSECTS (BAS);
177 1004 1
178 1005 1 !<BLF/PAGE>

```

```

! Define GLOBAL PSECTS same as code
! so short offsets to reach

```



```
180 1006 1 !+
181 1007 1 GLOBAL DISPATCH VECTORS (indexed by I/O statement type numbers):
182 1008 1 Connects the first level of abstraction (UPI) to the
183 1009 1 second level (UDF). Note: The comments down the
184 1010 1 side describe the I/O statement index (UPI level) into the
185 1011 1 dispatch table rather than the external routine contained in
186 1012 1 the entry (UDF level). The entries are the name of the
187 1013 1 User data formatters (UDF level = 2nd level of abstraction) -
188 1014 1 First letter: R = READ, W = WRITE; second letter: F = formatted,
189 1015 1 W = unformatted, L = list-directed.
190 1016 1 Declare as GLOBAL rather than GLOBAL BIND because
191 1017 1 BLISS doesn't allow BIND table = ... - table).
192 1018 1 -
193 1019 1 +
194 1020 1 Initialization of UDF level:
195 1021 1 -
196 1022 1
197 1023 1 GLOBAL
198 1024 1 BAS$$AA_UDF_PRO : VECTOR [ISB$K BASSTTYHI - ISB$K BASSTTYLO + 2,, SIGNED] INITIAL (
199 1025 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, ! I/O on closed unit
200 1026 1 ! I/O statement type:
201 1027 1 BAS$$UDF_WLO - BAS$$AA_UDF_PRO, BASIC Print
202 1028 1 BAS$$UDF_RLO - BAS$$AA_UDF_PRO, BASIC Linput
203 1029 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC PUT place holder
204 1030 1 BAS$$UDF_RLO - BAS$$AA_UDF_PRO, BASIC Input
205 1031 1 BAS$$UDF_WFO - BAS$$AA_UDF_PRO, BASIC Print Using
206 1032 1 BAS$$UDF_RLO - BAS$$AA_UDF_PRO, BASIC Input Line
207 1033 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC DELETE place holder
208 1034 1 BAS$$UDF_RMFO - BAS$$AA_UDF_PRO, BASIC Read memory formatted
209 1035 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC UPDATE place holder
210 1036 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC GET sequential
211 1037 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC RESTORE place holder
212 1038 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC SCRATCH place holder
213 1039 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC PUT relative place holder
214 1040 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC GET relative place holder
215 1041 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC FIND relative place holder
216 1042 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC UNLOCK place holder
217 1043 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC FREE place holder
218 1044 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC GET indexed place holder
219 1045 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC PUT indexed place holder
220 1046 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC MOVE FROM/MOVE TO place holder
221 1047 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC FIND indexed place holder
222 1048 1 BAS$$UDF_RLO - BAS$$AA_UDF_PRO, BASIC MAT INPUT
223 1049 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC RESTORE place holder
224 1050 1 BAS$$UDF_RLO - BAS$$AA_UDF_PRO, BASIC eventually MAT LINPUT
225 1051 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC FIND seq. place holder
226 1052 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC place holder
227 1053 1 BAS$$UDF_WLO - BAS$$AA_UDF_PRO, BASIC MAT PRINT
228 1054 1 BAS$$UDF_RMFO - BAS$$AA_UDF_PRO, BASIC MAT READ
229 1055 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC GET by RFA place holder
230 1056 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PRO, BASIC FIND by RFA place holder
231 1057 1 !<BLF/PAGE>
```

```
233 1058 1 !+
234 1059 1 ! Transmit a single I/O list element
235 1060 1 !-
236 1061 1
237 1062 1 GLOBAL
238 1063 1 BAS$$AA_UDF_PR1 : VECTOR [ISB$K BASSTTYHI - ISB$K BASSTTYLO + 2,, SIGNED] INITIAL (
239 1064 1 -BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, ! I/O on closed unit
240 1065 1 ! I/O statement type:
241 1066 1 BAS$$UDF_WL1 - BAS$$AA_UDF_PR1, BASIC Print
242 1067 1 BAS$$UDF_RL1 - BAS$$AA_UDF_PR1, BASIC Linput
243 1068 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC place holder for PUT
244 1069 1 BAS$$UDF_RLT - BAS$$AA_UDF_PR1, BASIC Input
245 1070 1 BAS$$UDF_WF1 - BAS$$AA_UDF_PR1, BASIC Print Using
246 1071 1 BAS$$UDF_RL1 - BAS$$AA_UDF_PR1, BASIC Input Line
247 1072 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC DELETE place holder
248 1073 1 BAS$$UDF_RMF1 - BAS$$AA_UDF_PR1, BASIC Read memory formatted
249 1074 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC UPDATE place holder
250 1075 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC GET seq. place holder
251 1076 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC RESTORE place holder
252 1077 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC SCRATCH place holder
253 1078 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC PUT relative place holder
254 1079 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC GET relative place holder
255 1080 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC FIND relative place holder
256 1081 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC UNLOCK place holder
257 1082 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC FREE place holder
258 1083 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC GET indexed place holder
259 1084 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC PUT indexed place holder
260 1085 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC MOVE FROM/MOVE TO place holder
261 1086 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC FIND indexed place holder
262 1087 1 BAS$$UDF_RLT - BAS$$AA_UDF_PR1, BASIC MAT INPUT
263 1088 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC RESTORE place holder
264 1089 1 BAS$$UDF_RLT - BAS$$AA_UDF_PR1, BASIC Mat Linput
265 1090 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC FIND seq. place holder
266 1091 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC place holder
267 1092 1 BAS$$UDF_WLT - BAS$$AA_UDF_PR1, BASIC MAT PRINT
268 1093 1 BAS$$UDF_RMF1 - BAS$$AA_UDF_PR1, BASIC MAT READ
269 1094 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC GET by RFA place holder
270 1095 1 BAS$$SIGDIS_ERR - BAS$$AA_UDF_PR1, BASIC FIND by RFA place holder
271 1096 1 !<BLF/PAGE>
```



```
273 1097 1 !+
274 1098 1 !- End I/O list entry points:
275 1099 1 !-
276 1100 1
277 1101 1 GLOBAL
278 1102 1 BAS$$AA_UDF_PR9 : VECTOR [ISB$K_BASSTTYHI - ISB$K_BASSTTYLO + 2,, SIGNED] INITIAL (
279 1103 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, ! I/O on closed unit
280 1104 1 ! I/O statement type:
281 1105 1 BAS$$UDF_WL9 - BAS$$AA_UDF_PR9, BASIC Print
282 1106 1 BAS$$UDF_RL9 - BAS$$AA_UDF_PR9, BASIC Linput
283 1107 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC place holder for PUT
284 1108 1 BAS$$UDF_RL9 - BAS$$AA_UDF_PR9, BASIC Input
285 1109 1 BAS$$UDF_WF9 - BAS$$AA_UDF_PR9, BASIC Print Using
286 1110 1 BAS$$UDF_RL9 - BAS$$AA_UDF_PR9, BASIC Input Line
287 1111 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC DELETE place holder
288 1112 1 BAS$$UDF_RMF9 - BAS$$AA_UDF_PR9, BASIC Read memory formatted
289 1113 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC UPDATE place holder
290 1114 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC GET seq. place holder
291 1115 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC RESTORE place holder
292 1116 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC SCRATCH place holder
293 1117 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC PUT relative place holder
294 1118 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC GET relative place holder
295 1119 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC FIND relative place holder
296 1120 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC UNLOCK place holder
297 1121 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC FREE place holder
298 1122 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC GET indexed place holder
299 1123 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC PUT indexed place holder
300 1124 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC MOVE FROM/MOVE TO place holder
301 1125 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC FIND indexed place holder
302 1126 1 BAS$$UDF_RL9 - BAS$$AA_UDF_PR9, BASIC MAT INPUT
303 1127 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC RESTORE place holder
304 1128 1 BAS$$UDF_RL9 - BAS$$AA_UDF_PR9, BASIC Mat Linput
305 1129 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC FIND seq. place holder
306 1130 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC place holder
307 1131 1 BAS$$UDF_WL9 - BAS$$AA_UDF_PR9, BASIC MAT PRINT
308 1132 1 BAS$$UDF_RMF9 - BAS$$AA_UDF_PR9, BASIC MAT READ
309 1133 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC GET by RFA place holder
310 1134 1 BAS$$SIGDIS_JSB - BAS$$AA_UDF_PR9, BASIC FIND by RFA place holder
311 1135 1 !<BLF/PAGE>
```

```
313 1136 1 +
314 1137 1 Dispatch tables to call record processing level of abstraction
315 1138 1 routines (REC = 3rd level). Used to connect 2nd level of
316 1139 1 abstraction (UDF) to third level of abstraction (REC).
317 1140 1 The dispatch tables are indexed by I/O statement type (1st
318 1141 1 level UPI.)
319 1142 1 Record processing routine names have the form:
320 1143 1 First letters: R = READ, W = WRITE));
321 1144 1 Second letters: S = sequential, D = direct, M = memory));
322 1145 1 third letters: F = formatted, U = unformatted, L = list-directed.
323 1146 1
324 1147 1 +
325 1148 1 Initialize entry points (read first record or setup
326 1149 1 output buffer).
327 1150 1
328 1151 1
329 1152 1 GLOBAL
330 1153 1 BAS$$AA_REC_PRO : VECTOR [ISB$K_BASSTTYHI - ISB$K_BASSTTYLO + 2., SIGNED] INITIAL (
331 1154 1 -BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, ! I/O to closed unit
332 1155 1 ! I/O statement type:
333 1156 1 BAS$$REC_WSLO - BAS$$AA_REC_PRO, BASIC Print
334 1157 1 BAS$$REC_RSLO - BAS$$AA_REC_PRO, BASIC Linput
335 1158 1 BAS$$REC_PSE - BAS$$AA_REC_PRO, BASIC PUT sequential
336 1159 1 BAS$$REC_RSLO - BAS$$AA_REC_PRO, BASIC Input
337 1160 1 BAS$$REC_WFO - BAS$$AA_REC_PRO, BASIC Print Using
338 1161 1 BAS$$REC_RSLO - BAS$$AA_REC_PRO, BASIC Input Line
339 1162 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC DELETE place holder
340 1163 1 BAS$$REC_RMFO - BAS$$AA_REC_PRO, BASIC Read memory formatted
341 1164 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC UPDATE place holder
342 1165 1 BAS$$REC_GSE - BAS$$AA_REC_PRO, BASIC GET sequential
343 1166 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC RESTORE place holder
344 1167 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC SCRATCH place holder
345 1168 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC PUT relative place holder
346 1169 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC GET relative place holder
347 1170 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC FIND relative place holder
348 1171 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC UNLOCK place holder
349 1172 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC FREE place holder
350 1173 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC GET indexed place holder
351 1174 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC PUT indexed place holder
352 1175 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC MOVE FROM/MOVE TO place holder
353 1176 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC FIND indexed place holder
354 1177 1 BAS$$REC_MINO - BAS$$AA_REC_PRO, BASIC MAT INPUT
355 1178 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC RESTORE indexed place holder
356 1179 1 BAS$$REC_RSLO - BAS$$AA_REC_PRO, BASIC Mat Linput
357 1180 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC FIND seq. place holder
358 1181 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC place holder
359 1182 1 BAS$$REC_WSLO - BAS$$AA_REC_PRO, BASIC MAT PRINT
360 1183 1 BAS$$REC_RMFO - BAS$$AA_REC_PRO, BASIC MAT READ
361 1184 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC GET by RFA place holder
362 1185 1 BAS$$SIGDIS_JSB - BAS$$AA_REC_PRO, BASIC FIND by RFA place holder
363 1186 1 !<BLF/PAGE>
```



```

365 1187 1 !+
366 1188 1 ! Intermediate transfer a record - read second and
367 1189 1 ! subsequent records for this I/O statement or write
368 1190 1 ! first and all but last record for this I/O statement.
369 1191 1 !-
370 1192 1
371 1193 1 GLOBAL
372 1194 1 BAS$$AA_REC PR1 : VECTOR [ISB$K BAS$T$YHI - ISB$K BAS$T$YLO + 2,, SIGNED] INITIAL (
373 1195 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, ! I/O to closed unit
374 1196 1 ! I/O statement type:
375 1197 1 BAS$$REC_WSL1 - BAS$$AA_REC PR1, BASIC Print
376 1198 1 BAS$$REC_RSL1 - BAS$$AA_REC PR1, BASIC Linput
377 1199 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC place holder for PUT
378 1200 1 BAS$$REC_RSL1 - BAS$$AA_REC PR1, BASIC Input
379 1201 1 BAS$$REC_WF1 - BAS$$AA_REC PR1, BASIC Print Using
380 1202 1 BAS$$REC_RSL1 - BAS$$AA_REC PR1, BASIC Input Line
381 1203 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC DELETE place holder
382 1204 1 BAS$$REC_RMF1 - BAS$$AA_REC PR1, BASIC Read memory formatted
383 1205 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC UPDATE place holder
384 1206 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC GET seq. place holder
385 1207 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC RESTORE place holder
386 1208 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC SCRATCH place holder
387 1209 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC PUT relative place holder
388 1210 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC GET relative place holder
389 1211 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC FIND relative place holder
390 1212 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC UNLOCK place holder
391 1213 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC FREE place holder
392 1214 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC GET indexed place holder
393 1215 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC PUT indexed place holder
394 1216 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC MOVE FROM/MOVE TO place holder
395 1217 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC FIND indexed place holder
396 1218 1 BAS$$REC_MIN1 - BAS$$AA_REC PR1, BASIC MAT INPUT
397 1219 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC RESTORE indexed place holder
398 1220 1 BAS$$REC_ML1 - BAS$$AA_REC PR1, BASIC eventually MAT LINPUT
399 1221 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC FIND seq. place holder
400 1222 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC place holder
401 1223 1 BAS$$REC_WSL1 - BAS$$AA_REC PR1, BASIC MAT PRINT
402 1224 1 BAS$$REC_MRE1 - BAS$$AA_REC PR1, BASIC MAT READ
403 1225 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC GET by RFA place holder
404 1226 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR1, BASIC FIND by RFA place holder
405 1227 1 !<BLF/PAGE>

```

```
407 1228 1 !+
408 1229 1 !- End of I/O list record processing
409 1230 1 !-
410 1231 1
411 1232 1 GLOBAL
412 1233 1 BAS$$AA_REC PR9 : VECTOR [ISB$K BAS$TYHI - ISB$K BAS$TYLO + 2,, SIGNED] INITIAL (
413 1234 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, ! I/O to closed unit
414 1235 1 ! I/O statement type:
415 1236 1 BAS$$REC_WSL9 - BAS$$AA_REC PR9, BASIC Print
416 1237 1 BAS$$REC_RSL9 - BAS$$AA_REC PR9, BASIC Linput
417 1238 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC place holder for PUT
418 1239 1 BAS$$REC_RSL9 - BAS$$AA_REC PR9, BASIC Input
419 1240 1 BAS$$REC_WF9 - BAS$$AA_REC PR9, BASIC Print Using
420 1241 1 BAS$$REC_RSL9 - BAS$$AA_REC PR9, BASIC Input Line
421 1242 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC DELETE place holder
422 1243 1 BAS$$REC_RMF9 - BAS$$AA_REC PR9, BASIC Read memory formatted
423 1244 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC UPDATE place holder
424 1245 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC GET seq. place holder
425 1246 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC RESTORE place holder
426 1247 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC SCRATCH place holder
427 1248 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC PUT relative place holder
428 1249 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC GET relative place holder
429 1250 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC FIND relative place holder
430 1251 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC UNLOCK place holder
431 1252 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC FREE place holder
432 1253 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC GET indexed place holder
433 1254 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC PUT indexed place holder
434 1255 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC MOVE FROM/MOVE TO place holder
435 1256 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC FIND indexed place holder
436 1257 1 BAS$$REC_RSL9 - BAS$$AA_REC PR9, BASIC MAT INPUT
437 1258 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC RESTORE indexed place holder
438 1259 1 BAS$$REC_RSL9 - BAS$$AA_REC PR9, BASIC Mat Linput
439 1260 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC FIND seq. place holder
440 1261 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC place holder
441 1262 1 BAS$$REC_MPR9 - BAS$$AA_REC PR9, BASIC MAT PRINT
442 1263 1 BAS$$REC_RMF9 - BAS$$AA_REC PR9, BASIC MAT READ
443 1264 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC GET by RFA place holder
444 1265 1 BAS$$SIGDIS_JSB - BAS$$AA_REC PR9, BASIC FIND by RFA place holder
445 1266 1 !+
446 1267 1 !- Return to normal PSECT definition
447 1268 1 !-
448 1269 1 DECLARE_PSECTS (BAS);
449 1270 1 !
```



```

451 1271 1 ROUTINE BAS$$SIGDIS_ERR : CALL_CCB NOVALUE = !
452 1272 1
453 1273 1 ++
454 1274 1 FUNCTIONAL DESCRIPTION:
455 1275 1
456 1276 1 Signal an error from the I/O dispatch process. The error code
457 1277 1 signalled depends on the statement type. One statement type is
458 1278 1 used by CLOSE to catch dispatches on a closed unit, which can
459 1279 1 happen if the CLOSE is done as part of recursive I/O. If the
460 1280 1 statement type is not the one used by CLOSE, we have an error
461 1281 1 in the RTL (an invalid statement type).
462 1282 1
463 1283 1 FORMAL PARAMETERS:
464 1284 1
465 1285 1 NONE
466 1286 1
467 1287 1 IMPLICIT INPUTS:
468 1288 1
469 1289 1 ISB$B_STTM_TYPE.rb.r Statement type of I/O statement
470 1290 1
471 1291 1 IMPLICIT OUTPUTS:
472 1292 1
473 1293 1 NONE
474 1294 1
475 1295 1 ROUTINE VALUE:
476 1296 1 COMPLETION CODES:
477 1297 1
478 1298 1 NONE
479 1299 1
480 1300 1 SIDE EFFECTS:
481 1301 1
482 1302 1 Signals OTSS$ IO_CONCLO if the LUB is not open, or
483 1303 1 OTSS$FATINTERR if it is.
484 1304 1
485 1305 1 --
486 1306 1
487 1307 2 BEGIN
488 1308 2
489 1309 2 EXTERNAL REGISTER
490 1310 2 CCB : REF BLOCK [, BYTE];
491 1311 2
492 1312 2 IF ( NOT .CCB [LUB$V_OPENED])
493 1313 2 THEN
494 1314 2 ++
495 1315 2 The file must have been closed with I/O still active on it.
496 1316 2 --
497 1317 2 LIB$STOP (OTSS$ IO_CONCLO)
498 1318 2 ELSE
499 1319 2 ++
500 1320 2 This must be an attempt to use an unimplemented feature. It represents
501 1321 2 an internal error in the OTS.
502 1322 2 --
503 1323 2 LIB$STOP (OTSS$FATINTERR);
504 1324 2
505 1325 2 0
506 1326 1 END;

```

BASSDISPATCH\_T  
1-021

```
B 16
16-Sep-1984 00:19:30 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 11:54:52 [BASRTL.SRC]BASDISPAT.B32;1
```

Page 12  
(9)

.TITLE BASSDISPATCH\_T  
.IDENT \1-021\

.PSECT \_BASSCODE, NOWRT, SHR, PIC, 2

[illegible]



<BASS\$SIGDIS JSB-BASS\$AA REC PRO>, -  
<BASS\$REC WSC-BASS\$AA REC PRO>, -  
<BASS\$REC RSLO-BASS\$AA REC PRO>, -  
<BASS\$REC PSE-BASS\$AA REC PRO>, -  
<BASS\$REC RSLO-BASS\$AX REC PRO>, -  
<BASS\$REC WFO-BASS\$AA REC PRO>, -  
<BASS\$REC RSLO-BASS\$AX REC PRO>, -  
<BASS\$SIGDIS JSB-BASS\$AA REC PRO>, -  
<BASS\$REC RMFO-BASS\$AA REC PRO>, -  
<BASS\$SIGDIS JSB-BASS\$AA REC PRO>, -  
<BASS\$REC GSE-BASS\$AA REC PRO>, -





[illegible]

```

. EXTRN OTSS FATINTERR, OTSS IO-CONCLO
. EXTRN LIB$$TOP, BASS$UDF RCO
. EXTRN BASS$UDF_RL1, BASS$UDF_RL9
. EXTRN BASS$UDF_WF0, BASS$UDF_WF1
. EXTRN BASS$UDF_WF9, BASS$UDF_WL0
. EXTRN BASS$UDF_WL1, BASS$UDF_WL9
. EXTRN BASS$UDF_RMF0, BASS$UDF_RMF1
. EXTRN BASS$UDF_RMF9, BASS$REC_RSLO
. EXTRN BASS$REC_RSL1, BASS$REC_RSL9
. EXTRN BASS$REC_WF0, BASS$REC_WF1
. EXTRN BASS$REC_WF9, BASS$REC_WSL0
. EXTRN BASS$REC_WSL1, BASS$REC_WSL9
. EXTRN BASS$REC_RMF0, BASS$REC_RMF1
. EXTRN BASS$REC_RMF9, BASS$REC_GSE
. EXTRN BASS$REC_PSE, BASS$REC_MPR9
. EXTRN BASS$REC_MIN0, BASS$REC_MIN1
. EXTRN BASS$REC_MIN9, BASS$REC_MLI1
. EXTRN BASS$REC_MRE1

```

	08	FC	AB	E8	00002	
		00000000G	8F	DD	00006	
			06	11	0000C	
		00000000G	8F	DD	0000E	1\$:
00000000G	00		01	FB	00014	2\$:
				04	0001B	

```

        .WORD      Save nothing
        BLBS       -4(CCB), 1$
        PUSHL      #OTS$-10_CONCLO
        BRB        2$
        PUSHL      #OTS$ FATINTERR
        CALLS      #1, LIB$STOP
        RET

```

• 1271  
• 1312  
• 1317  
•  
• 1323  
•  
• 1326

; Routine Size: 28 bytes, Routine Base: \_BAS\$CODE + 02E8

BAS\$DISPATCH\_T  
1-021

F 16  
16-Sep-1984 00:19:30  
14-Sep-1984 11:54:52

VAX-11 Bliss-32 V4.0-742  
[BASRTL.SRC]BASDISPAT.B32;1

Page 16  
(9)

```
508 1327 1 ROUTINE BAS$$SIGDIS_JSB : JSB_UDFO NOVALUE = !
509 1328 1
510 1329 1
511 1330 1 ++
512 1331 1 FUNCTIONAL DESCRIPTION:
513 1332 1 Signal an error from the I/O dispatch process. The error code
514 1333 1 signalled depends on the statement type. One statement type is
515 1334 1 used by CLOSE to catch dispatches on a closed unit, which can
516 1335 1 happen if the CLOSE is done as part of recursive I/O. If the
517 1336 1 statement type is not the one used by CLOSE, we have an error
518 1337 1 in the RTL (an invalid statement type).
519 1338 1
520 1339 1 FORMAL PARAMETERS:
521 1340 1
522 1341 1 NONE
523 1342 1
524 1343 1 IMPLICIT INPUTS:
525 1344 1
526 1345 1 ISB$B_STTM_TYPE.rb.r Statement type of I/O statement
527 1346 1
528 1347 1 IMPLICIT OUTPUTS:
529 1348 1
530 1349 1 NONE
531 1350 1
532 1351 1 ROUTINE VALUE:
533 1352 1 COMPLETION CODES:
534 1353 1
535 1354 1 NONE
536 1355 1
537 1356 1 SIDE EFFECTS:
538 1357 1
539 1358 1 Signals OTSS$IO_CONCLO if the LUB is not open, or
540 1359 1 OTSS$FATINTERR if it is.
541 1360 1
542 1361 1 --
543 1362 1
544 1363 2 BEGIN
545 1364 2
546 1365 2 EXTERNAL REGISTER
547 1366 2 CCB : REF BLOCK [, BYTE];
548 1367 2
549 1368 3 IF ( NOT .CCB [LUB$V_OPENED])
550 1369 2 THEN
551 1370 2 ++
552 1371 2 The file must have been closed with I/O still active on it.
553 1372 2 --
554 1373 2 LIB$STOP (OTSS$IO_CONCLO)
555 1374 2 ELSE
556 1375 2 ++
557 1376 2 This must be an attempt to use an unimplemented feature. It represents
558 1377 2 an internal error in the OTS.
559 1378 2 --
560 1379 2 LIB$STOP (OTSS$FATINTERR);
561 1380 2
562 1381 2 0
563 1382 1 END; !End of BAS$$SIGDIS_JSB
```



```

08      FC  AB  E8 00000 BAS$$SIGDIS JSB:
      00000000G 8F DD 00004 BLBS -4(CCB), 1$ : 1368
      00000000G 06 11 0000A PUSHL #OTS$ _10_CONCLO : 1373
      00000000G 8F DD 0000C 1$: BRB 2$ : 1379
00000000G 00 01 FB 00012 2$: CALLS #1, LIB$STOP : 1382
      05 00019 RSB

```

; Routine Size: 26 bytes, Routine Base: \_BAS\$CODE + 0304

```

: 564      1383 1 END !End of module
: 565      1384 1
: 566      1385 0 ELUDOM

```

# PSECT SUMMARY

Name	Bytes	Attributes
_BAS\$CODE	798	NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(2)

# COMMAND QUALIFIERS

```

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LISS$:BASDISPAT/OBJ=OBJ$:BASDISPAT MSRC$:BASDISPAT/UPDATE=(ENH$:BASDISPAT
: )

```

```

: Size: 54 code + 744 data bytes
: Run Time: 00:12.7
: Elapsed Time: 00:27.1
: Lines/CPU Min: 6558
: Lexemes/CPU-Min: 34456
: Memory Used: 135 pages
: Compilation Complete

```



0021 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY